

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ УКРАИНЫ

**ХАРЬКОВСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ
ГОРОДСКОГО ХОЗЯЙСТВА имени А. Н. БЕКЕТОВА**

Б. И. Погребняк, М. В. Булаенко

WEB-ТЕХНОЛОГИИ И WEB-ДИЗАЙН

КОНСПЕКТ ЛЕКЦИЙ

*(для студентов дневной и заочной форм обучения специальностей
122 – Компьютерные науки,
126 – Информационные системы и технологии
и 151 – Автоматизация и компьютерно-интегрированные технологии)*

**Харьков
ХНУГХ им. А. Н. Бекетова
2018**

Погребняк Б. И. Web-технологии и Web-дизайн : конспект лекций (для студентов дневной и заочной форм обучения специальностей 122 – Компьютерные науки, 126 – Информационные системы и технологии и 151 – Автоматизация и компьютерно-интегрированные технологии) / Б. И. Погребняк, М. В. Булаенко ; Харьков. нац. ун-т гор. хоз-ва им. А. Н. Бекетова. – Харьков : ХНУГХ им. А. Н. Бекетова, 2018. – 19 с.

Авторы:

канд. техн. наук, доц. Б. И. Погребняк,

канд. техн. наук, доц. М. В. Булаенко

Рецензент

А. Б. Костенко, кандидат физико-математических наук, доцент кафедры прикладной математики и информационных технологий (Харьковский национальный университет городского хозяйства имени А. Н. Бекетова)

Конспект лекций построен в соответствии с требованиями кредитно-модульной системы организации учебного процесса и согласован с ориентировочной структурой содержания учебной дисциплины, рекомендованной Европейской кредитно-трансферной системой (ECTS).

Рекомендовано кафедрой прикладной математики и информационных технологий, протокол № 16 от 17 мая 2017 г.

© Б. И. Погребняк, М. В. Булаенко, 2018

© ХНУГХ им. А. Н. Бекетова, 2018

ОГЛАВЛЕНИЕ

ЛЕКЦИЯ №1 ОСНОВЫ ВЕБ	4
1. Назначение языка HTML	4
2. Понятие Web-страницы и Web-сайта.....	5
3. История создания языка HTML.....	6
4. Структура HTML-документа	7
5. Инструментарий для создания HTML-документов.....	11
СПИСОК РЕКОМЕНДОВАННЫХ ИСТОЧНИКОВ	18

ЛЕКЦИЯ № 1

ОСНОВЫ ВЕБ

План:

1. Назначение языка HTML
2. Понятие Web-страницы и Web-сайта
3. История создания языка HTML
4. Структура HTML-документа
5. Инструментарий для создания HTML-документов

1 Назначение языка HTML

Наиболее востребованным на сегодняшний день сервисом Интернета является всемирная паутина – World Wide Web (WWW). Она «соткана» из множества отдельных Web-страниц, каждая из которых состоит из:

- 1) *содержания*;
- 2) *представления*;
- 3) *поведения*.

Содержание Web-страницы – это текст, графика, видео и аудио – одним словом, это информация, которая помещается на страницу. Для описания структуры этой информации служит язык *HTML (Hyper Text Markup Language* – язык гипертекстовой разметки). Представление описывает формат вывода этой информации. Именно представление делает Web-страницы такими привлекательными. Представление Web-страниц реализуется при помощи каскадных таблиц стилей (*Cascading Style Sheets – CSS*). Поведение – это динамическое изменение Web-страницы в зависимости от действий пользователя или каких-либо иных событий. Оно задается с помощью так называемых *сценариев (скриптов)* – программ, которые могут быть записаны прямо в коде Web-страниц. Такие программы создаются на одном из скриптовых языков. Чаще всего – на *JavaScript*.

Основой любой Web-страницы является ее содержание – без содержания страница существовать не может. Оно описывается (размечается) при помощи языка HTML. То есть, с помощью языка HTML описываются различные фрагменты документа (текст, рисунки, гиперссылки и т. д.), а так же их взаимное расположение.

Для просмотра документов, размеченных по правилам языка HTML, предназначены специальные программы – *браузеры (browsers)*. Они читают файл HTML-документа, форматируют его в виде Web-страницы и отображают на мониторе пользователя. Существует большое разнообразие программ-браузеров от различных производителей. Однако, на сегодняшний день, наиболее распространенными являются Internet Explorer корпорации Microsoft, а также Google Chrome разработанный в корпорации Google Inc, Mozilla Firefox от Mozilla Corporation и Opera от Opera Software.

2 Понятие Web-страницы и Web-сайта

Web-страницей обычно называют отдельный HTML-документ, размеченный при помощи языка HTML и предназначенный для публикации в сети Интернет. Набор таких Web-страниц, связанных общей тематикой, называют Web-сайтом.

Одна из этих страниц является основной (головной, стартовой, индексной и т. д.) и автоматически выдается на просмотр пользователю, указавшему в браузере только имя сайта. Все остальные страницы сайта, как правило, выдаются пользователю из основной при помощи гиперссылок.

Каждый отдельно взятый Web-сайт, как правило, кроме общей тематики, отличается еще и общим дизайном, т. е. общий вид и компоновка страниц. Это позволяет посетителям сайта легко на нем ориентироваться, так как страницы выполнены в едином стиле, на всех страницах одинаковая или похожая навигация, а меняется только их содержание.

3 История создания языка HTML

Начало развития языка HTML было положено выпускником Оксфордского университета, бакалавром в области физики, Тимом Бернерс-Ли (Tim Berners Lee). В 1989 году он выдвинул предложение о *Системе гипертекстовых документов* для создания внутренней информационной службы Европейского центра ядерных исследований (CERN). В 1990 году он назвал ее World Wide Web (на русский язык это можно перевести как Всемирная Паутина). Одной из составляющих этой системы был язык гипертекстовой разметки. Основные принципы языка HTML были заложены Тимом Бернерс-Ли в том же 1990 году, когда он работал над программой первого текстового Web-браузера. За основу им был взят язык для логического оформления текста SGML (Standard Generalized Markup Language), стандарт которого был принят Международной организацией по стандартизации (ISO) в 1986 году.

А первым (и долгое время единственным) графическим браузером в те далекие времена была программа Mosaic, разработанная, как и сама World Wide Web, в научном учреждении – Национальном центре суперкомпьютерных приложений США (National Center for Supercomputer Applications – NCSA) группой программистов возглавляемой Марком Андресеном в сентябре 1993 года. Быстроте, с которой новинка покорила виртуальное пространство, может позавидовать иной вирус – всего за год около двух миллионов пользователей установили Mosaic на свои компьютеры. Именно благодаря популярности браузера Mosaic язык HTML получил такое распространение.

В том же 1993 году появилась первая версия языка – HTML 1.0. Но поскольку она не была стандартизована, то разные производители браузеров стали предлагать свои варианты инструкций HTML, число которых все возрастало и возрастало. А для существования сети необходимо, чтобы авторы Web-страниц и производители браузеров пользовались одними и теми же соглашениями. Это и послужило причиной для начала совместной работы по созданию спецификации языка HTML. Такую работу взяла на себя

организация, называемая World Wide Web Consortium (сокращенно W3C). В ее задачу входило составление спецификации, отражающей текущий уровень развития возможностей языка с учетом разнообразных предложений компаний-разработчиков браузеров. Так, в ноябре 1995 года появилась спецификация HTML 2.0, призванная формализовать сложившуюся к концу 1994 года практику использования HTML. К тому времени новую версию языка HTML полностью поддерживало большинство браузеров. Последней версией языка, которая стала стандартом в 1999 году, является HTML 4.01. Следующая версия языка – HTML 5 – находится на стадии разработки. Однако большинство современных браузеров уже сейчас поддерживают его в том или ином виде.

4 Структура HTML-документа

HTML-документ представляет собой обыкновенный *текстовый файл* в формате ANSI ASCII, который состоит из двух типов объектов:

- 1) объекты, которые должны отображаться на Web-странице (текст, графика, мультимедиа и т. д.);
- 2) объекты, которые управляют расположением и отображением на Web-странице первых.

Объекты, которые управляют расположением и отображением на Web-страницах текста, графики и т. д., а сами на экране не отображаются, в языке HTML принято называть *тегами* (от английского слова *tag* – ярлык, признак). Теги вместе с объектами (фрагментами документа), отображением которых он управляет, называются *элементами* Web-страницы. Формальный синтаксис элемента представлен на рисунке 1.1.

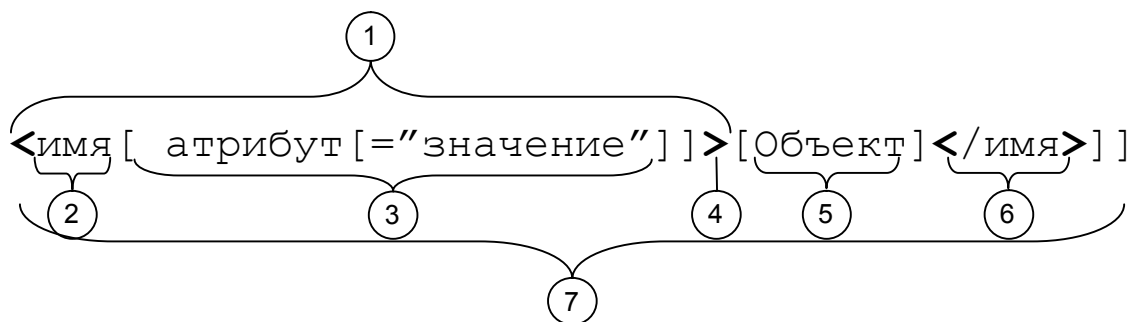



Рисунок 1.1 – Элемент HTML-документа

- | | |
|-----------------------|-----------------------|
| 1 открывающий тег | 5 отображаемый объект |
| 2 имя тега | 6 закрывающий тег |
| 3 атрибут | 7 элемент |
| 4 символ-ограничитель | |

 Составляющие в квадратных скобках «[]» являются не обязательными.

Для отделения тегов от других объектов HTML-документа они заключаются в символы-ограничители «<» и «>» (поз. 4. рис. 1.1), между которыми записываются *имя* тега (поз. 2. рис. 1.1) и, возможно, его *атрибуты* (поз. 3. рис. 1.1). Имя тега указывает на характер выполняемых им действий по разметке и форматированию, а атрибуты (параметры) уточняют способ выполнения этих действий. Многие атрибуты требуют указания их значений, которые следуют за знаком равенства, стоящим после имени атрибута. Если значение атрибута – одно слово или число, то его можно просто указать после знака равенства, не выделяя дополнительно. Все остальные значения необходимо заключать в одинарные «'» или двойные кавычки «"», особенно если они содержат несколько разделенных пробелами слов. Длина значения атрибута ограничена 1 024 символами. Некоторые атрибуты значений не имеют или могут быть записаны без них. В последнем случае они принимают некоторые значения по умолчанию. Атрибуты являются не обязательными составляющими тега и если они имеются, то отделяются от имени, и друг от друга, по крайней мере, одним пробелом. Порядок следования атрибутов тега произволен.

Большинство тегов языка HTML являются контейнерами, т. е. отображаемый объект (поз. 5. рис. 1.1) помещается между открывающим (поз. 1. рис. 1.1) и, соответствующим ему, закрывающим (поз. 6. рис. 1.1) тегами. Закрывающий тег имеет такие особенности:

1 имя закрывающего тега идентично имени открывающего;

1) перед именем закрывающего тега всегда ставится наклонная черта (прямой слеш) «/»;

2) закрывающий тег не имеет атрибутов (параметров).

Однако не все теги являются контейнерами – некоторые состоят только из открывающего, например, `
` – перевод строки. Такие элементы называются *пустыми*.

В языке HTML регистр символов в именах тегов и атрибутов не учитывается. Поэтому их можно записывать как строчными, так и прописными буквами, или их сочетанием. Например, правильно будут обработаны такие теги, как `
`, `
` и `
`. Однако, при записи значений атрибутов регистр символов следует учитывать.

Среди множества тегов языка HTML есть несколько *обязательных*, т. е. тех, которые всегда должны присутствовать в любом *действительном* HTML-документе. Их немного – всего четыре: `<html>`, `<head>`, `<title>` и `<body>` – вместе они создают структуру HTML-документа (Рис. 1.2).

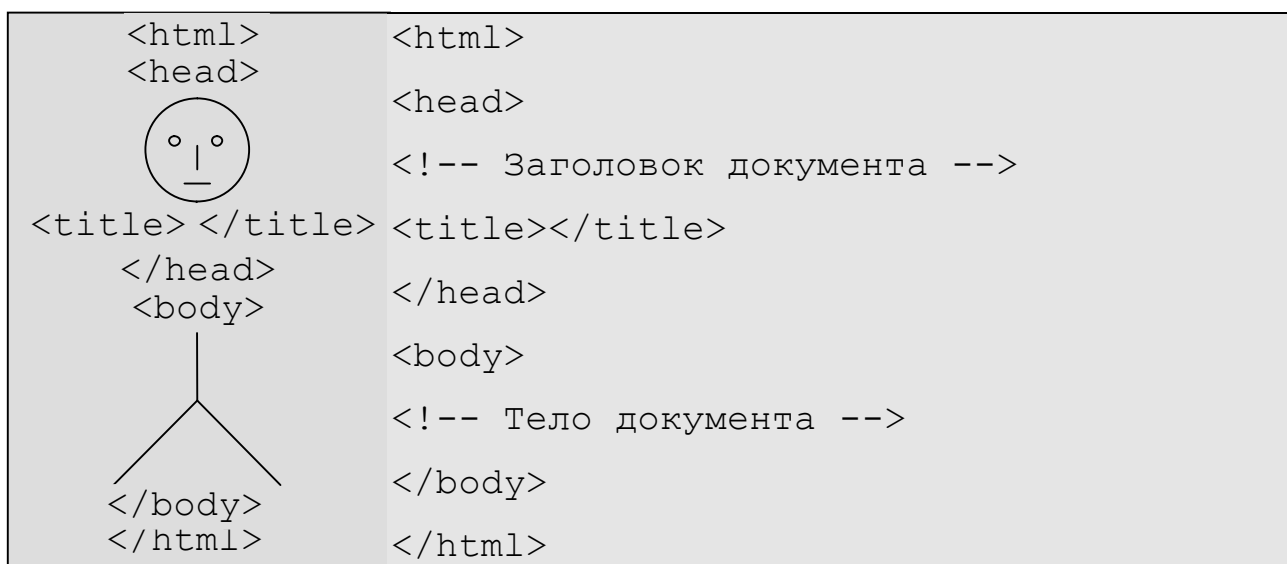


Рисунок 1.2 – Структура HTML-документа

Любой HTML-документ – это большой `html` элемент-контейнер. Его первым, открывающий, тегом должен быть `<html>` и, соответственно,

последним, закрывающим – `</html>`. Этот контейнер, в свою очередь, содержит два элемента:

1) *head* (заголовок), который оформляется при помощи тегов `<head>` и `</head>`,

2) *body* (тело), которое оформляется при помощи тегов `<body>` и `</body>`.

В заголовке HTML-документа содержится в основном служебная информация о нем, тогда как тело документа содержит непосредственно сам документ – текст, рисунки и прочие элементы, которые отображаются на экране. Элементы, находящиеся внутри заголовка, браузером не отображаются. Исключение составляет элемент `title` – название документа, которое отображается в строке заголовка браузера.

Поскольку большинство тегов языка HTML являются контейнерами, то помимо отображаемых объектов они могут содержать и другие теги, т. е. теги могут быть вложенными друг в друга. При этом необходимо соблюдать определенный порядок открывающих и закрывающих тегов: *тег, который был открыт первым, закрывается последним, а последний – первым*. То есть, такая последовательность тегов:

`<тэГ1><тэГ2> . . . </тэГ2></тэГ1>` – правильная,

а такая:

`<тэГ1><тэГ2> . . . </тэГ1></тэГ2>` – нет.

HTML-документ помимо тегов может содержать так же *комментарии*. Они ни как не влияют на отображение документа, а служат для пояснения отдельных его фрагментов. Например, если над документом работает несколько человек, или для того, чтобы через некоторое время было легче вспомнить и понять свой собственный текст. Комментарии так же иногда используются, когда необходимо временно исключить некоторый фрагмент кода из обработки, не удаляя его совсем. Для этого необходимо заключить такой фрагмент в комментарии, после чего браузером он будет игнорироваться.

Хотя текст комментариев и не отображается браузером, однако передается вместе с документом и вполне может быть просмотрен. Так происходит потому, что большинство браузеров предоставляют возможность просмотра исходного кода документа. Поэтому не следует включать в комментарии информацию, не предназначенную для чужих глаз.

Комментарии в языке HTML имеют следующий синтаксис:

```
<!-- Текст комментария -->
```

То есть, синтаксически это пустой элемент, имя которого «!--», а семантически (по смыслу) он, конечно же, отличается от любого тега.

5 Инструментарий для создания HTML-документов

Для создания любого документа нужна соответствующая программа, и Web-страницы не являются исключением. Существует различные способы создания таких страниц. Простейший из них состоит в том, что некоторые программы (такие, например, как Microsoft Word, Microsoft Excel, OpenOffice.org Writer, OpenOffice.org Calc, LibreOffice Writer, LibreOffice Calc и д. р.) «умеют» сохранять свои документы в формате .html. Для этого необходимо:

1) подготовить нужный документ;

2) выполнить команду **Файл ⇨ Сохранить как;**

3) в раскрывшемся окне диалога указать, что файл необходимо сохранить в формате HTML-документа.

Основные достоинства этого метода состоят в том, что Web-страницы создаются довольно быстро и просто, а так же совершенно не требуется знание языка HTML. Но на этом достоинства этого метода и заканчиваются – начинаются его недостатками. Наиболее существенные из них состоят в том, что страницы получаются «массового производства», их размеры в десятки раз превышают аналогичные, но созданные «вручную», изменять такие страницы в дальнейшем весьма проблематично. Соответственно, этот способ используется, в основном, в экстренных случаях – т. е. когда катастрофически не хватает

времени, а срочно необходимо один раз быстро создать несколько страниц, которые в дальнейшем не подлежат изменению.

Более совершенный способ создания Web-страниц состоит в использовании специальных программ – HTML-редакторов, которые бывают двух видов:

- 1) визуальные;
- 2) текстовые.

При использовании визуальных редакторов знания языка HTML является не обязательным. Они позволяют работать с Web-страницей «как она есть». То есть, пользователь редактирует и форматирует текст, вставляет рисунки, таблицы, как в обычном текстовом редакторе, а уж сама программа генерирует соответствующий HTML-код. Именно поэтому визуальные редакторы еще называют WYSIWYG-редакторами. Аббревиатура WYSIWYG расшифровывается как What You See Is What You Get – «что видишь, то и получишь». Следует отметить, что Microsoft Word (или OpenOffice.org Writer, или LibreOffice Writer), по существу, тоже является WYSIWYG-редактором, хотя и «прячут от пользователя» свои команды разметки и форматирования внутри текстового документа.

К наиболее распространенным визуальным редакторам относятся, например, Microsoft FrontPage, Macromedia DreamWeaver, HotDog от Sausage Software и некоторые другие. Они имеют мощные средства визуального редактирования, одновременно предоставляя доступ к созданному HTML-коду. Таким образом, работа в этих редакторах может состоять из двух шагов:

- 1 с начала визуальном режиме создается нужный HTML-документ;
- 2 а затем, в случае необходимости, можно произвести «более тонкую» его настройку, редактируя полученный на первом шаге код.

Использование визуальных HTML-редакторов существенно экономит время и силы разработчиков Web-страниц.

Однако следует заметить, что ни один визуальный редактор не совершенен, и все они, так или иначе, ограничены в своих возможностях.

Используя подобный редактор очень трудно создать что-либо оригинальное и эффектно смотрящееся, поскольку фантазия автора всегда будет ограничена рамками конкретной программы. А любому человеку всегда будет приятнее получить вещь, созданную мастером специально для него, чем приобрести аналог фабричного производства, который имеется дома у каждого второго. Поэтому для создания не «штампованных», оригинальных Web-страниц их код пишется руками при помощи текстовых HTML-редакторов, которые еще называются *редакторами тегов*.

Как правило, текстовые HTML-редакторы создаются небольшими фирмами или отдельными программистами, большинство из которых распространяется совершенно бесплатно. Среди таких редакторов есть совсем простые – например, WebCoder, созданный Дмитрием Захаровым. Редакторы тегов такого уровня, как правило, имеют различные функции, облегчающие написание кода. Так, например, при создании нового документа они позволяют быстро создавать заготовку будущей Web-страницы; по нажатию различных «горячих клавиш» и кнопок вставлять уже готовые конструкции (куски кода, спецсимволы и т. д.); автоматически вызывать браузер для просмотра редактируемого документа и т. п. Большинство из них так же поддерживает подсветку синтаксиса, т. е. HTML-код отображается различными цветами. Например, теги выделяются одним цветом, атрибуты – другим, их значения – третьим, а непосредственно текст страницы – четвертым. Это значительно упрощает чтение и редактирование кода – открыв документ, легко можно найти по цвету нужный элемент и отредактировать его.

Есть и более совершенные текстовые HTML-редакторы, такие, например, как Macromedia HomeSite, HTMLPad FisherMan, HTMLPad 2007, QuickHTML, NotePad++ (только для Windows), Quanta Plus (для UNIX-подобных операционных систем), Bluefish, Geany (есть версии для UNIX-подобных операционных систем, и для Windows) и многие, многие другие. Они позволяют не только создавать довольно сложные элементы, такие, например,

как таблицы, но и работать над проектами – наборами взаимосвязанных файлов, образующих сайт.

Между тем, поскольку HTML-документ является обычным текстовым файлом, то для его создания подойдет любой текстовый редактор. Простейший вариант – программа Блокнот, которая имеется у каждого пользователя, поскольку входит в стандартную поставку операционной системы Microsoft Windows. Она является наилучшим вариантом на начальном этапе освоения языка HTML – *с программы Блокнот когда-то начинали все создатели Web-сайтов*. Программа Блокнот всегда запустится по команде основного меню Windows **Пуск ⇒ Все программы ⇒ Стандартные ⇒ Блокнот**. Если при этом в нем набрать заготовку Web-страницы, представленную на Рис. 7.2, то она будет выглядеть следующим образом (рис. 1.3):

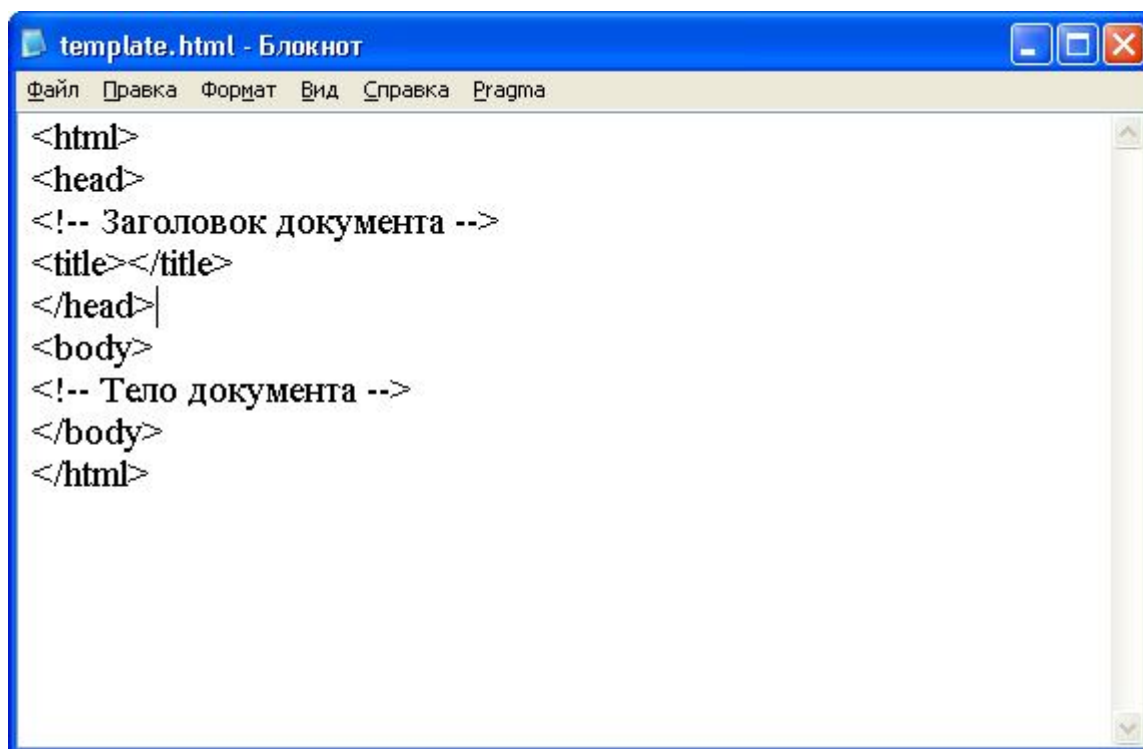



Рисунок 1.3 – Заготовка Web-страницы в программе **Блокнот**

 При первом сохранении HTML- документа в окне диалога **Сохранить** необходимо явно указать:

- 1) тип файла – **Текстовый документ**;
- 2) имя файла – латинскими буквами;
- 3) расширение имени файла – *.html* или *.htm*.

В дальнейшем необходимо привыкать к латинским буквам в именах файлов, поскольку не все компоненты Интернет адекватно работают с русскими символами.

Если теперь двойным щелчком открыть созданный HTML-документ в браузере, например, Microsoft Internet Explorer, то будет выведена пустая страница (рис. 1.4):

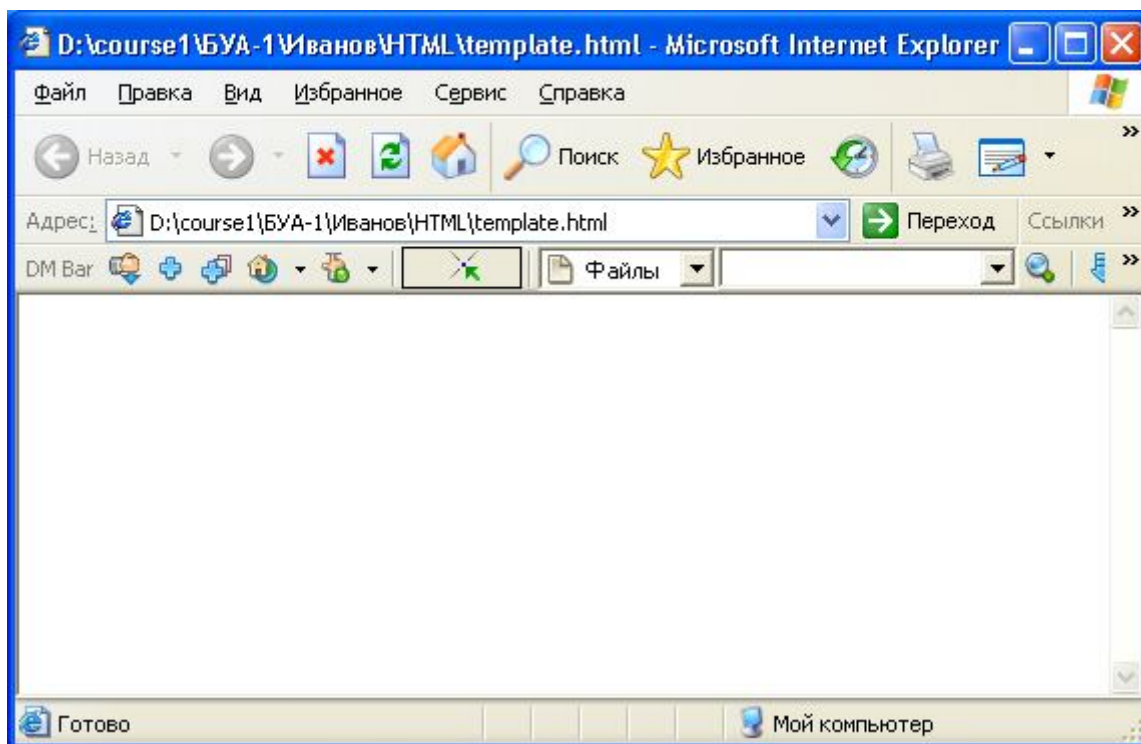


Рисунок 1.4 – Заготовка Web-страницы в браузере **Microsoft Internet Explorer**

Для того чтобы Web-страница приняла более информативный вид, необходимо в заготовку HTML-документа внести некоторые изменения. Например, между тегами <title> и </title> поместить текст «Мой первый документ HTML», а между <body> и </body> – «Всем огромный привет!». Тогда откорректированный текст HTML-документа должен выглядеть следующим образом:

```
<html>
<head>
<!-- Заголовок документа -->
<title>Мой первый документ HTML</title>
</head>
```

```
<body>
<!-- Тело документа -->
Всем огромный привет!
</body>
</html>
```

⚠ Для того, чтобы увидеть внесенные в HTML-документ изменения необходимо:

- 1) сохранить его в программе Блокнот (выполнив, например, команду **Файл** ⇒ **Сохранить**);
- 2) обновить содержание окна браузера (щелкнув по кнопке **Обновить**).

В результате откорректированный HTML-документ должен выглядеть как на рисунке 1.5.

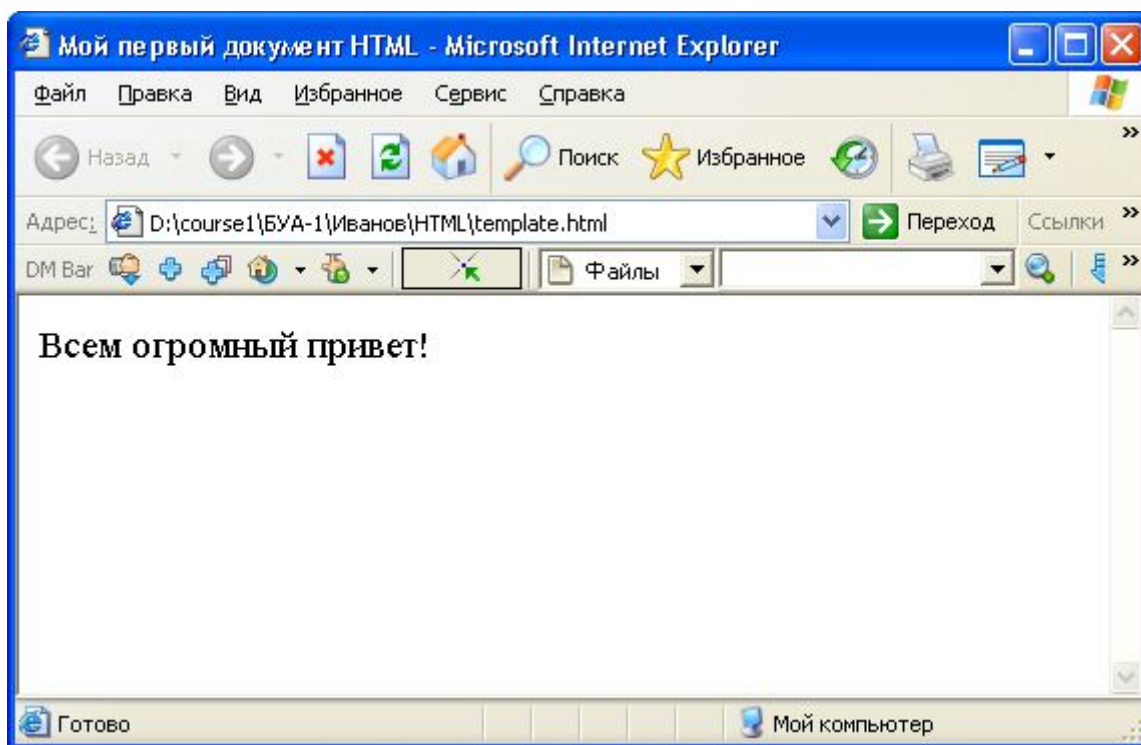


Рисунок 1.5 – Первый HTML-документ

Приведенные выше текст Web-страницы был разбит на строки только лишь с целью лучшей читабельности и восприятия человеком. Большинство современных браузеров воспринимают код HTML-файлов как одну длинную строку, пропуская при этом все лишние пробелы, символы конца строк и

табуляции. То есть, «с точки зрения браузера» этот текст выглядит следующим образом:

```
<html><head><!-- Заголовок документа --><title>Мой первый  
документ HTML</title></head><body><!-- Тело документа --  
>Всем огромный привет!</body></html>
```

Если этот текст так же (без разбивки на строки) набрать в Блокноте, то внешний вид Web-страницы в окне браузера при этом не изменится.

СПИСОК РЕКОМЕНДОВАННЫХ ИСТОЧНИКОВ

1. Дронов В. А. HTML 5, CSS 3 и Web 2.0. Разработка современных Web-сайтов / В. А Дронов. – СПб. : БХВ-Петербург, 2011. – 416 с. : ил.
2. Коржинский С. Н. Настольная книга Web-мастера : эффективное применение HTML, CSS, JavaScript / С. Н. Коржинский. – Издание второе, исправленное и дополненное. – М. : Издательский торговый дом «КноРус», 2000. – 320 с.
3. Матросов А. А. HTML 4.0 / А. А. Матросов, А. О. Сергеев, М. П. Чаунин. – СПб: БХВ, 2004. – 672 с. : ил.
4. Алленова Наталья : учебник по Html (хтмл) для чайников [Электронный ресурс]. – Режим доступа : <http://postroika.ru/html/>.
5. Городулин Владимир : HTML-справочник [Электронный ресурс]. – Режим доступа : <http://html.manual.ru/>.
6. Климов Александр : HTML в примерах [Электронный ресурс]. – Режим доступа : <http://webmaster.alexanderklimov.ru>.

Навчальне видання

ПОГРЕБНЯК Борис Іванович,
БУЛАСНКО Марина Володимирівна

WEB-ТЕХНОЛОГІЇ ТА WEB-ДИЗАЙН

КОНСПЕКТ ЛЕКЦІЙ

*(для студентів денної та заочної форм навчання спеціальностей
122 – Комп'ютерні науки, 126 – Інформаційні системи і технології
та 151 – Автоматизація і комп'ютерно-інтегровані технології)*

(Рос. мовою)

Відповідальний за випуск *О. Б. Костенко*

За авторською редакцією

Комп'ютерне верстання *І. В. Волосожарова*

План 2017, поз. 254Л

Підп. до друку 20.03.17. Формат 60 × 84/16.

Друк на ризографі. Ум. друк. арк. 0,8

Тираж 50 пр. Зам. № .

Видавець і виготовлювач:

Харківський національний університет
міського господарства імені О. М. Бекетова,
вул. Маршала Бажанова, 17, Харків, 61002
Електронна адреса: rectorat@kname.edu.ua

Свідоцтво суб'єкта видавничої справи:

ДК № 5328 від 11.04.2017.